

Almudena Perona Requena

21-06-2011

DOCK 6

¿Como trabaja Dock?

- 1) Preparación del Receptor y de los ligandos:
 - Añadir protones
 - Parámetros de vdW
 - Cargas
- 2) Calcular la superficie accesible al solvente de la proteína
- 3) Crear una imagen negativa de las características de la superficie del centro activo usando esferas
- 4) Calcular la energía de grid (cada punto de la grid contiene un valor de van der Waals y una carga para esa porción del espacio)
- 5) Docking: Ajustar el ligando a las esferas y puntuarlo contra la grid

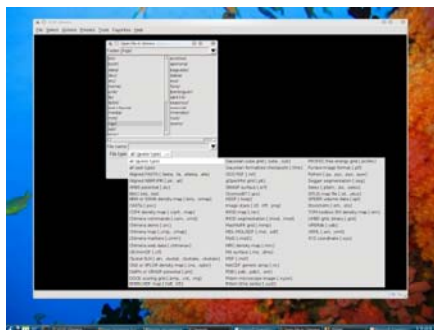
1.-Chimera: preparación del Receptor y de los ligandos

• Dock no posee interfaz gráfica, para la preparación de proteína y ligandos usa Chimera.

CHIMERA

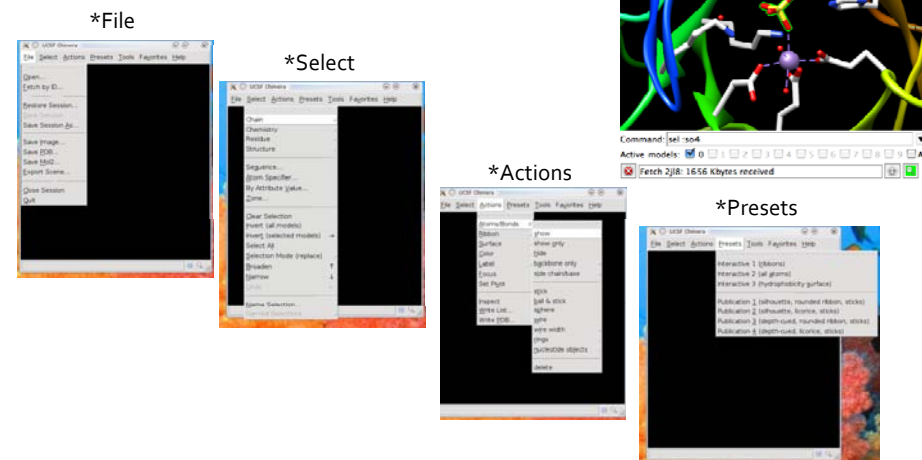
Es un programa interactivo para ver y trabajar de forma gráfica con las moléculas (receptores y ligandos).

- Puede usarse en distintos sistemas operativos: Linux, Windows y Mac
- Es gratuito
- Lee varios tipos de formato de ficheros de entrada



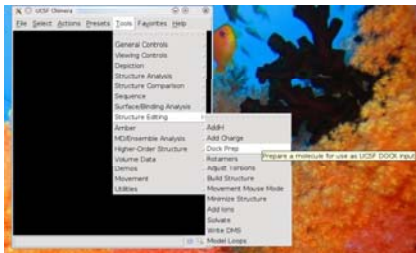
1.- Chimera

Tiene una barra horizontal de "menu" en la parte superior

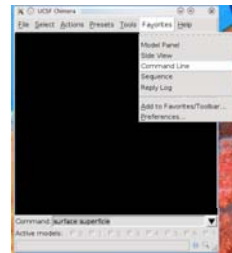


1.- Chimera

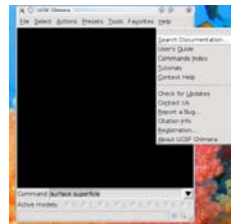
*Tools



*Favorites



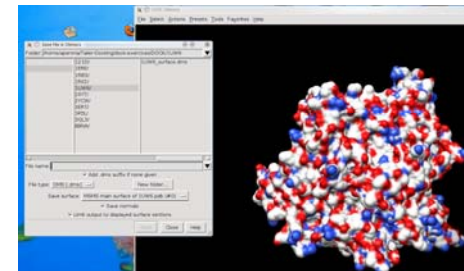
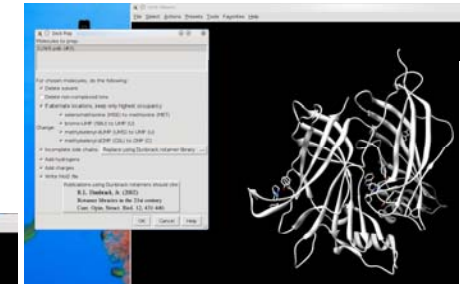
*Help



Todas las ordenes pueden ser ejecutadas también por línea de comando.

1 y 2.- Chimera: Dock Prep y Superficie

1) Añadir protones , Parámetros de vdW y Cargas



2) Calcular la superficie accesible al solvente de la proteína

3.- DOCK: Esferas

- Generación de esferas para rellenar el centro activo de la proteína
 - Están definidas por las coordenadas de centro (*orientación del ligado*) y por su radio (*clustering*)
 - El centro de las esferas corresponde a las posiciones de los átomos del ligando (se limita el numero de orientaciones posibles en el centro activo).
 - Están en contacto con la superficie de la proteína pero no traspasan las molécula
- *Orientación del ligando*: Distancia esfera - esfera es comparable a la distancia átomo - átomo en el ligando. Se hace coincidir pares esfera - átomo de modo que:

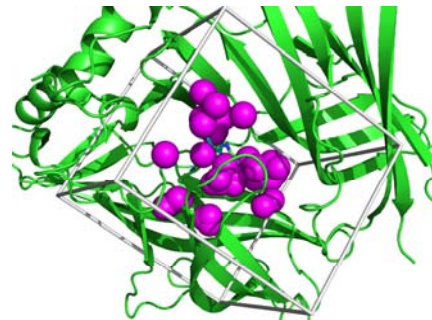
la esfera i se empareja con el átomo I si y solo si para cada esfera j en el conjunto y para cada átomo J en el conjunto :

$$|d_{ij} - d_{Ij}| < \epsilon$$

d_{ij} =distancia entre la esfera ij

d_{Ij} =distancia entre los átomos IJ

ϵ (*epsilon*) = es un valor definido por el usuario



3.- DOCK: Esferas

CARACTERÍSTICAS:

■ Puntuación Química de las esferas (*Chemical matching*): Se asocian propiedades química a colores de las esferas. Las parejas de átomos-esferas deben tener colores complementarios .

Las propiedades químicas son: Donador de puente de hidrógeno, Aceptor de puente de hidrógeno, hidrófobo, electro-positivo, electro-negativo, neutro, etc..

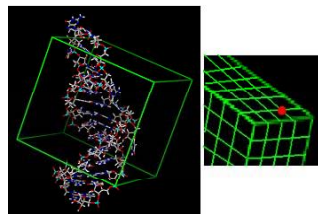
■ Esfera como Puntos Críticos (filtro): obliga al ligando a ocupar ese área en el centro activo.

■ Filtro de Choque (disminución del tiempo de calculo): comprobación de que el ligando no ocupa un espacio que ya está ocupado por el receptor. Si existen muchos choques esa orientación del ligando es eliminada.

4.- DOCK: Grid

■ Generación de una caja alrededor del centro activo: localización y centro de la grid.

- El centro de la grid se sitúa en el centro de masas de las esferas elegidas
- Se amplía la distancia elegida en 6 direcciones del espacio



■ Cálculo de la energía de grid para cada sitio activo del receptor. Cada punto de grid guarda la información de la energía de vdW y la carga para ese área del espacio.

- Crea los ficheros necesarios para puntuar en DOCK de dos formas: Contactos y Energía
- Al realizar el docking cada función de puntuación (**scoring function**) se aplica independientemente y se escriben dos ficheros de salida por separado: *.cnt (contactos) y *.nrg (energía)
- "grid de choque" (*bump grid*): identifica si un átomo del ligando va a tener severos choques estéricos con un átomo del receptor (fichero *.bmp).

4.- DOCK: Generación de la Grid

■ Vamos a usar la grid basada en la función de *scoring* de energía compuesta por términos que tiene en cuenta las energías de interacción de van der Waals y electrostática.

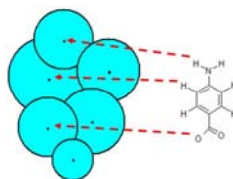
$$E = \sum_{i=1}^{lig} \sum_{j=1}^{rec} \left(\frac{A_{ij}}{r_{ij}^a} - \frac{B_{ij}}{r_{ij}^b} + 332 \frac{q_i q_j}{D r_{ij}} \right)$$

■ Se trata de una suma doble de los átomos del ligando (*i*) y del receptor (*j*).

5.- Docking Rígido

- Docking de un ligando rígido en una proteína rígida.
- Orientación: El algoritmo de docking intenta encajar o hacer coincidir los centros de los átomos pesados con las esferas siguiendo los siguientes pasos:

- 1) Generación de *nodos* o parejas átomo pesado - centro esfera
- 2) Etiquetarlos como coincidencia si las *longitudes de borde* (distancia euclídea entre átomos o centros de esferas) son equivalentes
- 3) Extensión de las coincidencias añadiendo más nodos
- 4) Generación exhaustiva de coincidencias
- 5) Crear una matriz de transformación (usando las coincidencias) que mueva a la molécula entera

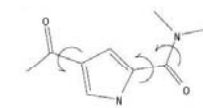


- Optimización energética usando un minimizador para optimizar la interacción entre el ligando y el receptor (durante la minimización el ligando puede ser flexible mientras que el receptor no).
- La puntuación final. La mejor pose generada del proceso de orientación y de minimización.

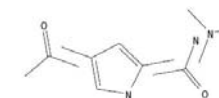
6.- Docking de Ligando flexible

- Ligando flexible, su estructura se "reorganiza" en respuesta del receptor.
- Identificación de la parte rígida de la estructura (*anchor*) y después las partes flexibles.
flex_definition_file parameter: contiene información acerca del tipo de átomo y el entorno químico de los átomos unidos además de información de las posibles posiciones de torsión.

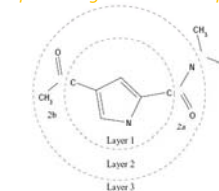
parte rígida de la estructura



Identificación de segmentos no solapados



Capas de segmentos no solapados



- Localización de cada enlace flexible y división de la molécula en segmentos rígidos.
- Redivisión en segmentos no solapables concéntricos.
- Orientación, puntuación o minimización de la parte rígida
- Unión sucesiva de los segmentos para buscar la conformación
- Optimización de la energía de interacción entre receptor-ligando con un minimizador.

7.- Scoring

Tipos de función de puntuación (*scoring*) que usa DOCK para elegir orientaciones y moléculas

- Una puntuación principal maneja todas las funciones que usa DOCK:
 - Primera función de scoring: Orientación rígida, crecimiento de la molécula y minimización.
 - La segunda función de scoring: Minimización final, puntuación y ordenación de las moléculas.
- **Bump Filter:** Filtro previo a la puntuación, descarta aquellas orientaciones y pasos de crecimiento en los que el ligando se solapa con la proteína
- **Puntuación (score) de Contactos:** Suma del número de contactos entre los átomos pesados del ligando y del receptor. Da una idea de la complementariedad de forma.
- **Puntuación (score) basada en la GRID:** Se basa en los términos de no enlace del campo de fuerzas de la mecánica molecular.
- **Puntuación (score) DOCK 3.5:** Variante de la anterior. Calcula la desolvatación del ligando y las interacciones estéricas y electrostáticas (on un mapa de potencial electrostático ESP) entre receptor y ligando.

7.- Scoring

- **Puntuación (score) continua:** Evalúa el complejo ligando-receptor sin calcular una grid. Se usan parámetros de *score* que proporciona DOCK.
- **Puntuación (score) Zou GB/SA:** Es un algoritmo muy rápido (modelo de energía libre *pairwise*) para calcular afinidad de unión del ligando. Se obtienen resultados similares a los basados en grid con menos esfuerzo computacional.
- **Puntuación (score) Hawkins GB/SA:** Implementación del método de Área de Superficie Generalizado de Born de Mecánica Molecular MM-GBSA.
Interacción total receptor-ligando se representa por términos coulombicos y de energía de Lennard jones (MM) mas el cambio en la solvatación (GBSA)
$$\Delta\text{GBSA} = \text{GBSA complejo} - (\text{GBSA receptor} + \text{GBSA ligando})$$
- **Puntuación (score) PB/SA:** Aplica la herramienta electrostática ZAP (OpenEye). ZAP soluciona la ecuación de Poisson-Boltzmann (PB) (relacionado con la distribución del campo dieléctrico y las cargas de la molécula)
- **Puntuación (score) AMBER:** Usa las herramientas de AMBER, mecánica molecular, solvatación implícita, simulación de dinámica molecular, flexibilidad del receptor y minimizador de gradientes conjugados. La preparación de los input es mas complicada y aumenta la carga computacional.

8.- Cribado Virtual (VS)

- VS en Dock equivale a hacer n veces dockings flexibles sencillos (n = número de moléculas). Es necesario hacer uso de herramientas informáticas como "scripts"

"Un *script* (cuya traducción literal es 'guion') o **archivo de órdenes** o **archivo de procesamiento por lotes** es un *programa* usualmente simple, que por lo regular se almacena en un archivo de texto plano. El uso habitual de los scripts es realizar diversas tareas como combinar componentes, interactuar con el sistema operativo o con el usuario"

- Generación de la grid de centro activo del receptor
- Crear un sistema de carpetas:
 - cada carpeta contiene el ligando con el que se hace el docking
 - ejecutar el programa que crea los ficheros de entrada necesarios para hacer el cálculo (*Lig_setup.sh*)
- Crear los ficheros de entrada "input" para hacer el docking flexible (*setup_flex-dock.sh*)
- Ejecutar el docking para n moléculas de forma automática (*flex-dock_VS.sh*)
- Obtener los resultados de cada molécula que están incluidos cada uno en su carpeta correspondiente. (*get_total_results.sh*)

8.-VS DUD

- DUD: Base de datos de *decoys* (*señuelos*). Se usa para probar algoritmos de docking
- Posee 2950 compuestos activos para 40 proteínas
- Por cada compuesto activo hay 36 "decoys": poseen las mismas propiedades químicas (peso molecular, LogP calculado...) pero distinta topología
- VS del antagonista del receptor de estrógenos:
 - Receptor : 2ERT.pdb
 - Ligando del cristal: OHT
 - Ligandos (1): 39
 - Decoys (0): 1448

8.-VS scripts

- Lig_setup.sh : Crea una carpeta por cada ligando y llama al programa que genera los ficheros *.in

```
#!/bin/bash

for i in `seq 1 39`; do
    mkdir -p "/ligs/mol$i"
    mv "ligs/mol$i.mol2" "ligs/mol$i/"
    ../flex-dock_VS.sh rec "mol$i" /data/acortes/er_antagonist > setup_dock
    cd ../
done
```

▪ numero de ligandos , se laman *mol{n}?.mol2*

▪ crea una carpeta para cada ligando

▪ introduce el ligando en la carpeta

▪ llama al script flex-dock_VS.sh que genera los "input"

7.-VS scripts

- setup_flex-dock.sh : programa que genera el fichero flex.in para cada ligando

```
#!/bin/bash

DOCK6_PATH=/data/ortizg/software/Linux/dock6_64/

REC_BASE_NAME=$1
LIG_BASE_NAME=$2
VS_PATH=$3

echo "ligand_atom_file"
echo receptor_site_file
echo grid_score_grid_prefix
echo "vdw_defn_file"
echo "flex_defn_file"
echo "flex_drive_file"

echo "limit_max_ligands"
skip_molecule
read_mol_solvation
calculate_rmsd
use_database_filter
orient_ligand
automated_matching
max_orientations
critical_points
chemical_matching
use_ligand_spheres
use_internal_energy
internal_energy_rep_exp
flexible_ligand
min_anchor_size
.....

$VS_PATH/$REC_BASE_NAME/_selected_spheres.sph >> "$REC_BASE_NAME".flex.in
$DOCK6_PATH/parameters/vdw_AMBER_parm99.defn >> "$REC_BASE_NAME".flex.in
$DOCK6_PATH/parameters/flex.defn >> "$REC_BASE_NAME".flex.in
$DOCK6_PATH/parameters/flex_drive.tbl >> "$REC_BASE_NAME".flex.in

$LIG_BASE_NAME.mol2 >> "$REC_BASE_NAME".flex.in
$VS_PATH/$REC_BASE_NAME/_grid >> "$REC_BASE_NAME".flex.in
```

▪ nombre del ligando

▪ centro activo

▪ grid

▪ parámetros

8.-VS scripts

- flex-dock_VS.sh : programa que manda ejecutar el docking para cada ligando

```
#!/bin/bash

DOCK6_PATH=/data/ortizg/software/Linux/dock6_64/
REC_BASE_NAME=$1
cd "/data/acortes/er_antagonist/ligands/ligs/mol$2"
echo "Performing flexible docking ..."
$DOCK6_PATH/bin/dock6 -i "$REC_BASE_NAME".flex.in -t -o dock6_result1
```

▪ carpeta donde está el programa de docking

▪ carpeta donde está el ligando

▪ ejecución de dock6

8.-VS scripts

- get_results_AC.sh : programa que de cada carpeta de cada ligando obtiene el valor de la energía de docking

```
#!/bin/bash

for i in `seq 1 39`; do
    score=`grep "Grid Score" "ligs/mol$i/dock6_result" | egrep -v "Initializing|Parameters" | tail -1 | cut -d ":" -f2 | bc`
    echo "$i;$score"
done
```

- get_total_results.sh : llama al programa que de cada carpeta de cada ligando o decoy obtiene el valor de la energía de docking y genera una lista ordenada.

```
#!/bin/bash

cd ligands
./get_results_AUC.sh > ../total
cd ../decoys
./get_results_AUC.sh >> ../total
cd ../
grep ";-;" total | sed 's/;/ /' | sort -k2g > total_order
```